Final Exam

RBS, Discrete Structures

2020-04-23

## Problem 1

By we denote the set of all positive integers between and . This is our *universe* in which we define several subsets:

**(A)** Express using the sets (using set union , set intersection , set complement operations).  
**(B)** Express using the sets in a similar way.  
**(C)** Find - the size of the set .  
**(D)** Find - the size of the set .

## Problem 2

Let and be sets with sizes and and .

Calculate the largest and the smallest possible values for each of the following set sizes:

**(A)** .  
**(B)** .  
**(C)** - the powerset of a powerset of .  
**(D)** - the symmetric difference of the sets and .

## Problem 3

Consider the following recurrent sequence:

Assume that is another sequence satisfying the recurrence rule

(The first two members are not known.)

**(A)** Write the first members of this sequence ().  
**(B)** Write the characteristic equation for this sequence.  
**(C)** Write the general expression for an arbirary sequence satisfying the recurrent expression as a sum of two geometric progressions (you can leave unknown coefficients in your answer; just explain which ones they are).  
**(D)** Write the formula to compute (that would satisfy the initial conditions and ).

## Problem 4

Consider this code snippet in Python:

n = 1000  
sum = 0  
for i in range(1, n\*n+1):  
 for j in range(1,i+1):  
 sum += i % j

And a similar one in R:

n <- 1000  
sum <- 0  
for (i in 1:(n\*n)) {  
 for (j in 1:i) {  
 sum <- sum + i %% j  
 }  
}

**(A)** Explain in human language what this algorithm does.  
**(B)** Denote by the number of times the variable sum is incremented. Write the Big-O-Notation for . Find a function such that is in . (If there are multiple functions, pick the one with the slowest growth.)  
**(C)** Express the function precisely - how many times sum is incremented in terms of variable .

## Problem 5

Let be the set of all positive divisors of the number (including and itself).  
**(A)** What is the multiplication of all numbers in the set ?  
**(B)** Express this number as the product of prime powers.

## Problem 6

Define the following binary relationship on the set of integer numbers :  
We say that (numbers are in the relation ) iff

|  |  |  |
| --- | --- | --- |
| Item | Statement | True or False? |
| **(A)** | is reflexive |  |
| **(B)** | is symmetric |  |
| **(C)** | is antisymmetric |  |
| **(D)** | is transitive |  |
| **(E)** | iff |  |

For all items where you answered FALSE, specify a counterexample (values for some numbers that would make the condition true, but the conclusion false). If the statement was true, write “none”.

**(A)** counterexample \_\_\_

**(B)** counterexample \_\_\_

**(C)** counterexample \_\_\_

**(D)** counterexample \_\_\_

**(E)** counterexample \_\_\_

## Problem 7

Four people each has his own hat. After the meeting they leave their building in a hurry, everyone grabs some hat at random so that all permutations of the hats have equal probabilities.

Let the random variable denote the number of hats that were picked up correctly. (For example, if the hat assignment is this: , then , because two people got their own hats.)

**(A)** Find - the expected value of .  
**(B)** Find - the variance of .

## Problem 8

There was a crooked man who had a crooked 1 euro coin. On lucky days it would flip the *heads* with probability , and the *tails* with probability , but on unlucky days it was the opposite (, but ). There were equal probabilities of for lucky and unlucky days.

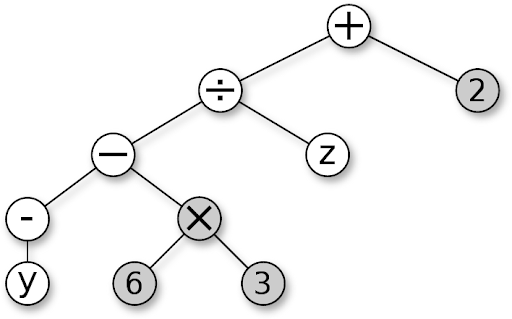
One morning he flipped the coin times and altogether got three *heads* and two *tails*.

Let us introduce the following events:

* (evidence): Five coin tosses result in three *heads* and two *tails*.
* (hypothesis): The current day is lucky.

**(A)** Find - the conditional probability of given that the day is lucky.  
**(B)** Find - the probability that the day is lucky and happens.  
**(C)** Find - the conditional probability of given that the day is not lucky.  
**(D)** Find - the probability that the day is unlucky and happens.  
**(E)** Find - as the sum of two probabilities ( happened on a lucky day and also happened on unlucky day).  
**(F)** Find the conditional probability - the likelyhood that the croocked man has a lucky day, given that the event has happened.

## Problem 9

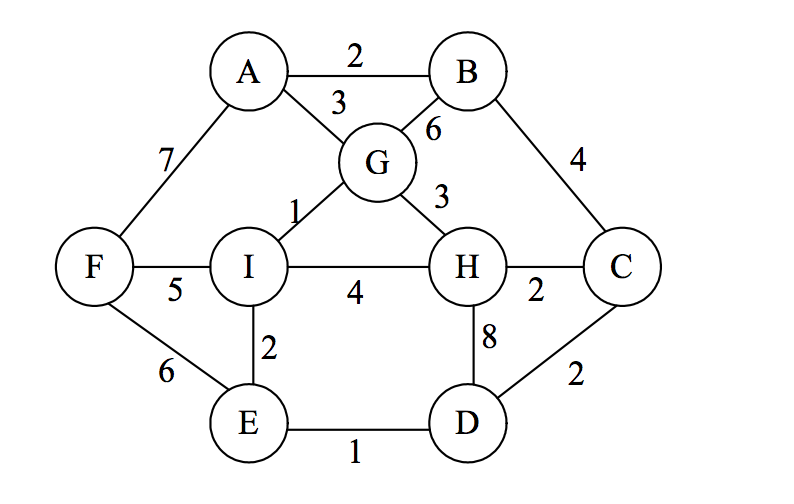
  
*Figure: A syntax tree for an expression*

The syntax tree describes an algebraic expression (please note the difference between the unary minus that flips the value of the variable and the binary minus that subtracts the two subexpressions: and ).

**(A)** Write the preorder DFS traversal of this tree.  
**(B)** Write the inorder DFS traversal of this tree.  
**(C)** Write the postorder DFS traversal of this tree.

*Note.* In all answers denote the unary minus with the tilde sign , but the regular/binary minus with .

## Problem 10

  
*Figure: Graph with 9 vertices*

Run the Prim’s algorithm on the following weighted graph, start growing the tree from the vertex .

|  |  |
| --- | --- |
| Step | Newly Added Edge |
| **Step 1** |  |
| **Step 2** |  |
| **Step 3** |  |
| **Step 4** |  |
| **Step 5** |  |
| **Step 6** |  |
| **Step 7** |  |
| **Step 8** |  |

What is the total weight of the obtained Minimum Spanning Tree? \_\_\_\_\_\_

# Answers

## Problem 1

**(A)** (Boolean OR means set union)  
**(B)** (Boolean and means set intersection; negation means set complement)  
**(C)** (principle of inclusion-exclusion).  
**(D)** is all odd numbers and also four even numbers divisible by (). The total is .

## Problem 2

In all the answers the largest and the smallest value are equal, because we know exactly how the two sets intersect; how many elements belong to just one of the sets , , and how many elements belong to the both sets.

**(A)** (the principle of inclusion-exclusion).  
**(B)** (Cartesian product has size that is the product of all participant sets: one can combine three elements from the sets , and in this many ways).  
**(C)** (the number of elements in the powerset of any set can be obtained by raising to the power ).  
**(D)** (we remove the common elements from both and ).

## Problem 3

**(A)** ,  
,  
,  
,  
,  
,  
.

**(B)** The characteristic equation is obtained, if we try to find in the form of a geometric progression :  
 or  
.  
It has two roots: , .

**(C)** The general form of the expression for any iterative sequence satisfying the relationship is as follows:

where are two constants that depend on the two initial values of the sequence .

**(D)** We need to solve a system of two equations, to ensure that the formula has correct values for and . We get the following system:

Substitute into the second equation. We get that and . We also get that . Therefore the exact formula to calculate the sequence is this:

This actually works, if we plug in the values calculated in **(A)** for .

## Problem 4

**(A)** The algorithm takes all numbers from to and divides them by all the smaller numbers , and adds up all the obtained remainders.

**(C)** The outer loop is repeated times. The inner loop is repeated times. This is an arithmetic progression. The sum of an arithmetic progression is the arithmetic mean of the first and the last member multiplied by the number of members:

**(B)** is in . Therefore we can take . We can pick another that is multiplied by some nonzero constant (such as or or anything else - that also counts as a valid answer).  
Certainly, is also in for any , but the function is the slowest growing.

## Problem 5

**(A)** If expressed as a product of two positive integers , one of the divisors or would be smaller than , and the other one would be bigger. We can easily list all the ways to express as a product of two integers:

and there are no other factorizations, since all the divisors less than are already listed.  
Multiplying them all together would give

**(B)** As a product of prime factors:

## Problem 6

|  |  |  |
| --- | --- | --- |
| Item | Statement | True or False? |
| **(A)** | is reflexive | TRUE |
| **(B)** | is symmetric | TRUE |
| **(C)** | is antisymmetric | FALSE |
| **(D)** | is transitive | TRUE |
| **(E)** | iff | FALSE |

**(A)** Counterexample: None  
**(B)** Counterexample: None  
**(C)** Consider counterexample , .  
While it is true that and , nevertheless .  
**(D)** Counterexample: None  
**(E)** Counterexample is same as in **(C)**: , .

## Problem 7

* For of permutations (all hats stay in place),
* For permutations (it is not possible for exactly three hats to stay in place, because then the 4th hat also returns to its owner),
* For of permutations (there are ways how to pick hats that stay in place; and the remaining two hats can switch places only in one way),
* For of permutations (there are ways how to pick hat that stays in place; and the remaining three hats can rotate in two ways).
* For the remaining permutations (no hats stay in place).

**(A)** . This means that the expected number of hats that stay in place is exactly .  
**(B)** For all permutations, subtract the value from every hat experiment outcome. To make addition faster, we group the terms by their value (one value , six values and so on):

Therefore, (variance also equals , but the unit of measurement is not hats but ``hats squared’’).

## Problem 8

**(A)** is the outcome of the Binomial distribution: There are coin-toss experiments; the probability of success for any single experiment is (since we know that the day is lucky and hypothesis holds). Therefore,

**(B)** , since (the *a priori* probability of a lucky day is exactly ).

**(C)** is the outcome of the Binomial distribution: Again, there are coin-toss experiments, but now the probability of a single experiment is just . Therefore,

**(D)** .

**(E)** We can compute as the sum of two mutually incompatible events: event can happen either on a lucky day or on an unlucky day:

**(F)** Use Bayes formula:

Bayes formula is intuitive: It shows the proportion of the subcase (i.e. event hapens on a lucky day) out of the whole probability (i.e. event happens either on a lucky or unlucky day).

## Problem 9

**(A)** ,  
**(B)** ,  
**(C)** .

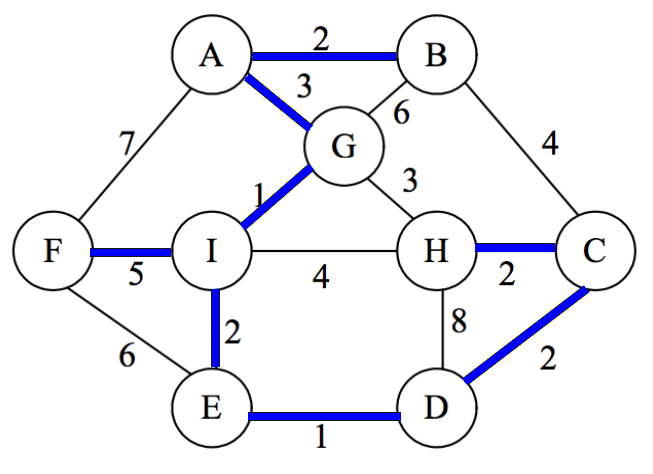
Note: In inorder traversal (**(B)**) we first visit the first subtree (e.g., ), and only then the parent node (e.g., unary minus ). See (Rosen2019, p.811).

## Problem 10

We start from vertex . At every step we grow the tree by a single edge (so that it stays connected and the newly added edge has the smallest possible weight).

|  |  |
| --- | --- |
| Step | Newly Added Edge |
| **Step 1** | , |
| **Step 2** | , |
| **Step 3** | , |
| **Step 4** | , |
| **Step 5** | , |
| **Step 6** | , |
| **Step 7** | , |
| **Step 8** | , |

The total weight of all added edges (same as the total weight of the MST) is .

  
*Figure: MST edges shown in blue*